# Robot Programming #10

## Motors

Dept. of Mech. Robotics and Energy Eng.
Dongguk University

# RC Servo Motor

- A RC servo motor is a small rotary position control device, used for example in radio-controlled cars and aero planes to position controllers such as steering, elevators and rudders.
- It was made to control the position of the steering wheel in a RC car and a RC airplane.
- Recently, it is popularly used in robots.

# Running the RC servo motor

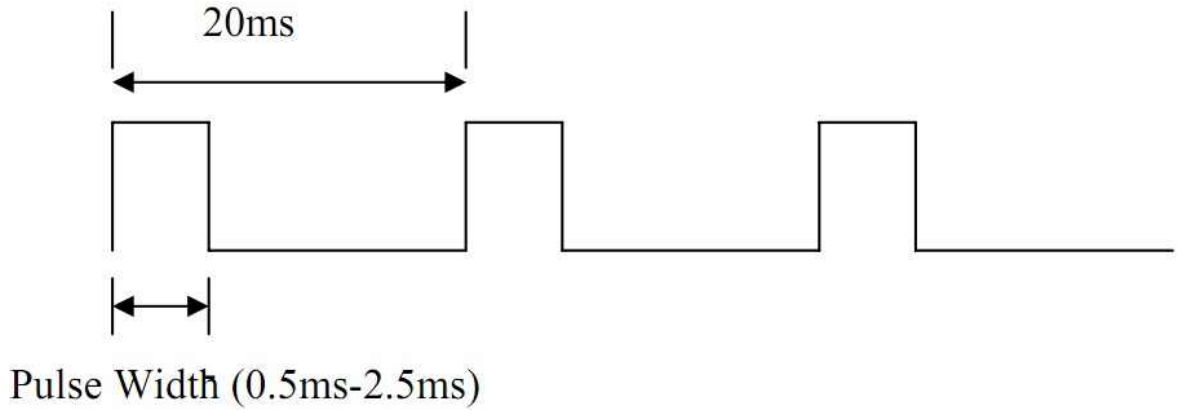
- A servo motor consists of several main parts, the motor and gearbox, a position sensor(potentiometer), an error amplifier, motor driver, and a circuit to decode the requested position.
- The RC servo motor has three wires(Power, Ground, Control signal).
- Power(red) should be connected to the positive of the power supply.
- Ground(black) is connected to the negative of the power supply and the ground of the mbed.
- Control signal(yellow) is to be connected to the output of the mbed.

# Running the RC servo motor

- The servo shaft can be positioned to specific angular positions by sending the servo a PWM signal
- As long as the modulated signal exists on the input line, the servo will maintain the angular position of the shaft.
- As the modulated signal changes, the angular position of the shaft changes.

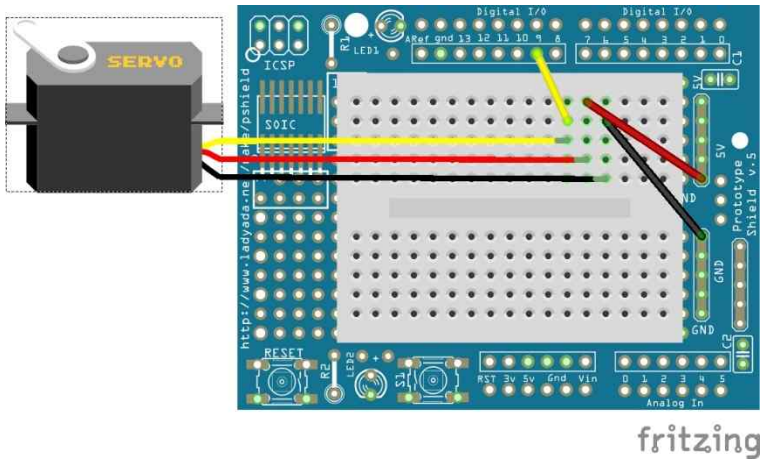
# Controlling the servo motor

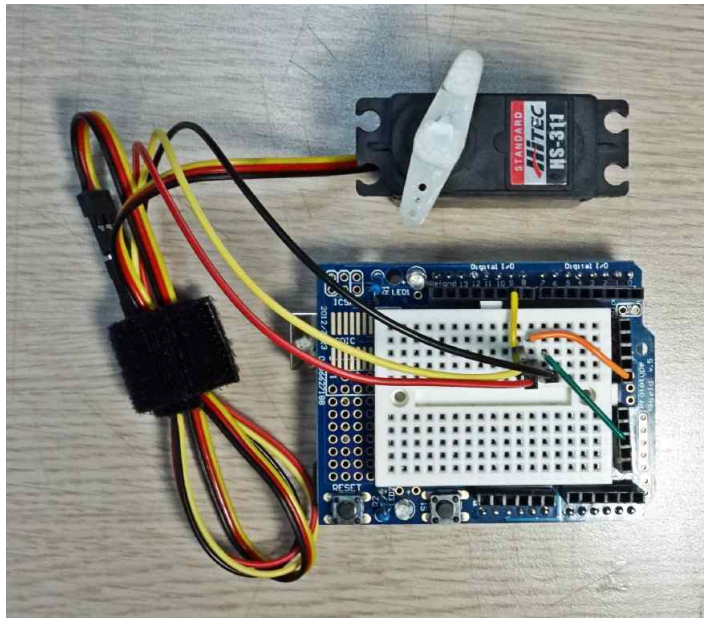
- The servo is controlled by a series of pulses, wherein the length of the pulse indicates the position to take.
- Increasing the pulse width by 10 μs results in about a degree of movement on the output shaft.
- The servo can hold the position when the signal is fed with 20 ms period.



# Controlling servo position

- Connect the servo control to the pin 9.
- The servo requires a higher current than the USB standard can provide, and so it is essential that you power the servo using an external power supply.



# Controlling the servo position

```
#include <Servo.h>
int servoPin = 9;
Servo servo;
int angle = 0; // servo position in degrees
void setup()
{
    servo.attach(servoPin);
}
void loop()
{
for(angle = 0; angle < 180; angle++) // from 0 to 180 degrees
  {
    servo.write(angle);
    delay(15);
  }
for(angle = 180; angle > 0; angle--) // from 180 to 0 degrees
  {
    servo.write(angle);
    delay(15);
  }
}
```

# A Closer Look

- We will use the library for the operation of the RC servo motor.

```
#include <Servo.h>
```

- Define servo variable.

```
Servo servo;
```

- Connect servo to the pin.

```
servo.attach(servoPin);
```

- Rotate the servo to the desired angle.

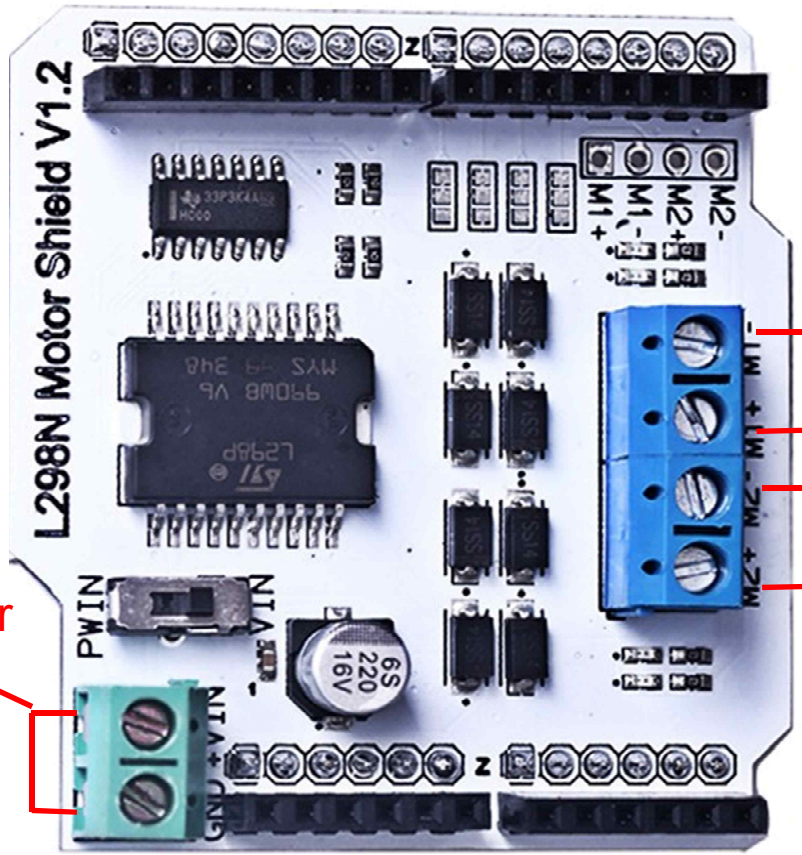
```
servo.write(angle);
```

# DC Motor

- A DC motor is structurally very simple.
- Running a DC motor is easier than RC servo motor.
- However, controlling the speed of a DC motor is not an easy task.
- In the case of RC servo motor, the angular position of the RC servo motor is determined by the pulse signal.
- However, if we want a DC motor to do the same task, we need extra devices and driver.
- The simplest method is to use the motor shield for Arduino.

# Motor Shield

- Motors are conected internally to pins 4-7.




External Power for Driver
(4.8V-35V)

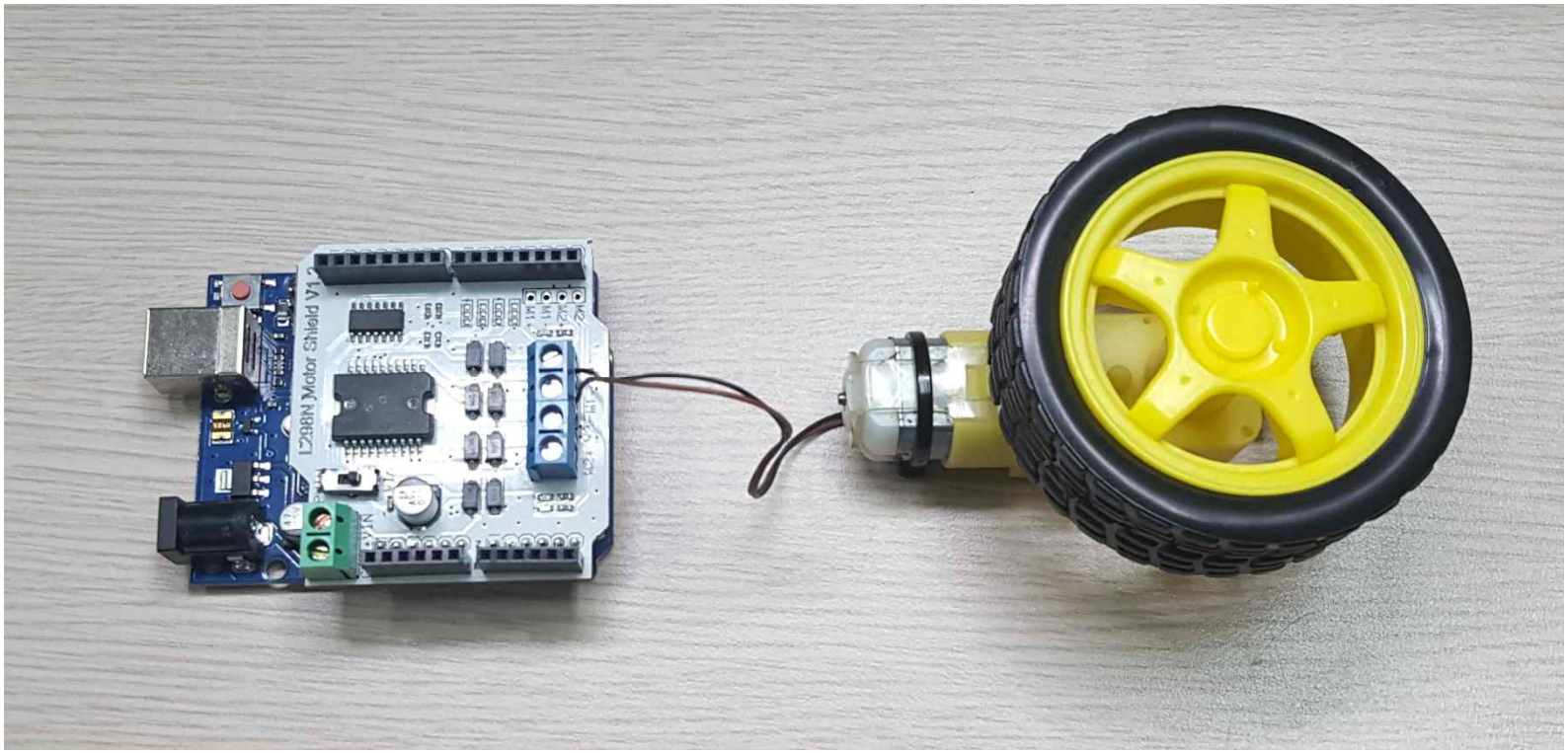
# Motor Shield

- Plug in the motor shield to Arduino and connect DC motors to it.

| Pin | Function |
|---|---|
| Digital 4 | Motor 1 Direction control |
| Digital 5 | Motor 1 PWM control |
| Digital 6 | Motor 2 Direction control |
| Digital 7 | Motor 2 PWM control |

# Motor Shield

- Plug in the motor shield to Arduino and connect DC motors to it.



# Motor Driver Test

- Write the code to test the driver and DC motors.

```
int E1 = 4;       // pin for M1 direction
int M1 = 5;       // pin for M1 speed

void setup(){
  Serial.begin(9600);
  pinMode(M1, OUTPUT);
}
void loop(){
  Serial.println("Stop");
  digitalWrite(E1,HIGH);
  analogWrite(M1, 0);
  delay(500);

  Serial.println("Forward");
  digitalWrite(E1,HIGH);
  analogWrite(M1, 255);
  delay(3000);

  Serial.println("Backward");
  digitalWrite(E1,LOW);
  analogWrite(M1, 255);
  delay(3000);

  Serial.println("Stop");
  digitalWrite(E1,HIGH);
  analogWrite(M1, 0);
  delay(500);
}
```